

数组

目录

◆ 一维数组

◆ 二维数组

- ▶ 数组是用来存储一组相同类型数据的数据结构。可以通过整型下标（index）访问数组中的每一个值。在Java中，数组也是独立的对象，有自身的方法，不再单纯是变量的集合。

- ▶ 一个数组中所有的元素应该是同一类型；
- ▶ 数组中的元素是有顺序的；
- ▶ 数组中的一个元素通过数组名和数组下标确定。



一维数组



翡翠教育
EMERALD EDUCATION

▶ 生成数组步骤（三步）

▶ 声明

▶ 创建

▶ 赋值



一维数组

▶ 数组的声明

- ▶ 类型标识符 数组名[] ; 或 类型标识符[] 数组名 ;
- ▶ `int intArray [];` - - ([] 不是指长度可变)
- ▶ `int[] intArray; int abc[]; double[] example2;`
- ▶ 数组的长度不一定在声明时指定。严格来说是在创建时，按照数组的长度创建对应大小的内存空间。



一维数组

▶ 在声明的同时创建

```
int intArray [ ] = new int [10] ;
```

▶ 直接赋值

```
int[ ] intArray = { 1, 2, 3, 4, 5 } ;
```



一维数组

► 数组的创建

► 先声明后创建

- `int intArray [] ;`
- `intArray = new int [10] ;`



一维数组

▶ 数组中元素的赋值

▶ 直接赋值

```
int[ ] intArray = { 1, 2, 3, 4, 5 } ;
```

▶ 先创建后赋值

```
int intArray[ ] = new int [10] ;  
for(int i=0;i<10;i++){  
    intArray[i] = i;  
}
```

- ▶ 一维数组元素的引用格式如下：
 - ▶ 数组名[数组下标]
 - ▶ 数组下标的取值范围是0 ~ (数组长度-1)，下标值可以是整数型常量或整数型变量表达式。



一维数组

▶ 冒泡排序(口诀)

- ▶ N个数字来排序，两两相比大靠前，
- ▶ 外层循环N-1，内层循环N-1-i.

```
public class BubbleSort {  
    public void sort(int[] args){  
        int time1 = 0,time2 = 0;  
        for (int i = 0; i < args.length - 1; i++) {  
            ++time1;  
            for (int j = i + 1; j < args.length; j++) {  
                ++time2;  
                int temp;
```

一维数组

```
int temp;
if (args[i] > args[j]) {
    temp = args[j];
    args[j] = args[i];
    args[i] = temp;
}
}
}
System.out.println();
System.out.println("排序后 : ");
for (int n : args) {
    System.out.print(n + " ,");
}
```

▶ 小练习：

- ▶ 创建一个方法，用来查询一个整型值在一个整型数组中的下标值是多少？



一维数组

小练习：

有如下规律的一组数字，求该组数字的第137位。

1 1 2 3 5

8 13 21 34 55



二维数组

- ▶ 严格来说，Java中只有一维数组，而不存在“二维数组”。在Java语言中，二维数组实际上可以看成是其每个数组元素都是一维数组的一个由一维数组组成的一维数组。这里面最根本原因，是因为计算机存储器的编址是一维结构的，二维数组的结构即是一堆顺序排列的一维数组。

▶ 二维数组的声明需给出两对方括号：

```
类型说明符 数组名[ ][ ];      int arr[ ][ ];  
类型说明符[ ][ ] 数组名;      int [ ][ ] arr;
```

- ▶ 类型说明符可以是Java的基本类型、类或接口，也可以是自定义的类型的对象；数组名是用户遵循标识符命名规则给出的一个标识符；两个方括号中前面的方括号表示行，后面的方括号表示列。

▶ 二维数组的创建

- ▶ 二维数组声明同样也是为数组命名和指定数据类型的，并不为数组元素分配内存。只有创建之后才能为其分配存储空间。
- ▶ 二维数组的创建也分为直接指定初始值和用new操作符先创建再赋值两种方式。

▶ 直接指定初值

- ▶ 在数组声明时对数据元素赋初值就是用指定的初值对数组初始化。

```
int[ ][ ] arr1={{3, -9,6},{8,0,1},{11,9,8} };
```

- ▶ 声明并初始化数组arr1，它有3个元素，每个元素又都是有3个元素的一维数组。

▶ 创建固定长度的数组

- ▶ 数组声明后，可用两种格式中的任意一种来初始化二维数组。
- ▶ 数组名=new 类型说明符[数组长度][]；
- ▶ 数组名=new 类型说明符[数组长度][数组长度]；



二维数组

- ▶ **创建不固定长度一维数组元素的二维数组**
 - ▶ 在初始化二维数组时可以只指定数组的行数而不给出数组的列数，每一行的长度由二维数组引用时决定。但不能只指定列数而不指定行数。

```
int[ ][ ] arr = new int[3][ ];  
arr[0] = new int[1] ;  
arr[1] = new int[3] ;  
arr[2] = new int[2] ;
```



二维数组

▶ 二维数组的赋值

- ▶ 除了在声明创建时直接赋值外，二维数组同样可以通过for循环赋值。

```
int arr[3][4];  
    for(int i=0;i<arr.length;i++){//遍历数组的每一  
行  
        for(int j=0;j<arr[i].length;j++){  
            //遍历第i行数组的每一列  
            arr[i][j]=i+j;  
        }  
    }
```

谢 谢

